BACANAL : Balades Aléatoires Courtes pour ANAlyses Lexicales Application à la substitution lexicale

Yann Desalle, Emmanuel Navarro, Yannick Chudy, Pierre Magistry, Bruno Gaume

BACANAL : Balades Aléatoires Courtes pour ANAlyses Lexicales
Application à la substitution lexicale

Yann Desalle[1] Emmanuel Navarro[2] Yannick Chudy Pierre Magistry[3, 4] Bruno Gaume[5]
(1) ATILF, CNRS, Université de Lorraine
(2) IRIT, CNRS, Université de Toulouse
(3) Graduate Institute of Linguistics, National Taiwan University
(4) LPL, CNRS, Aix Marseille Université
(5) CLLE-ERSS, CNRS, Université de Toulouse
yann.desalle@gmail.com, navarro@irit.fr, ychudy@gmail.com, pmagistry@gmail.com,
gaume@univ-tlse2.fr

Résumé. Nous proposons ici des méthodes de désambiguisation sémantique par substition lexicale pour la tâche 1 de l'atelier SemDis2014. Les méthodes exposées dans ce papier sont toutes bâties à partir de balades aléatoires courtes dans des graphes unipartis ou bipartis construits sur diverses ressources. Certaines de ces méthodes n'utilisent que des graphes construits automatiquement à partir de corpus (*méthodes non supervisées*), d'autres utilisent des graphes construits à partir de ressources produites « à la main » par des lexicographes ou par les foules (*méthodes supervisées*).

Abstract. In this paper, we propose word sense disambiguation methods based on lexical substitution and used for the task 1 of the SemDis2014 workshop. This methods are run by using short random walks on unipartite networks or bipartite networks. Some of these methods only use graphs automatically built from corpora (*unsurpervised methods*), others also use graphs built from handcraft resources filled by lexicographers or by the crowds (*supervised methods*).

Mots-clés : désambiguisation sémantique, substition lexicale, réseaux lexicaux, balades aléatoires courtes.

Keywords: word sense disambiguation, lexical substitution, lexical networks, short random walks.

1 Introduction

Depuis l'article de (McCarthy, 2002), la tâche de substitution lexicale s'est répandue : elle est de plus en plus utilisée dans des tâches telles que, par exemple, la désambuiguisation sémantique (McCarthy & Navigli, 2009) ou l'interprétation automatique de métaphores (voir (Desalle *et al.*, 2009; Desalle, 2012) pour le français et (Shutova, 2010; Shutova *et al.*, 2012) pour l'anglais).

Nous proposons ici des méthodes de désambiguisation sémantique par substitution lexicale pour la tâche 1 de l'atelier SemDis2014 [1], adaptation pour le français de la tâche 10 de SemEval2007 (McCarthy & Navigli, 2007) à la suite de (Van de Cruys *et al.*, 2011). La particularité de cette tâche est de ne pas fournir à l'avance l'inventaire des substituts possibles à ordonner en fonction des contextes d'occurrence de l'item à désambiguïser : cette inventaire est à déterminer en amont par le système.

Les méthodes de désambiguisation par substitution lexicale développées jusqu'à aujourd'hui pour cette tâche se répartissent en deux catégories : (a) d'une part les méthodes qui s'appuient sur des ressources lexicales construites à la main telles que WordNet (Fellbaum, 1998), le Rodget's Thesaurus, le Macquarie Thesaurus etc. pour déterminer l'inventaire des candidats-substituts (à l'aide de filtres du type « synonymes seulement ») avant de les ordonner par des méthodes non-supervisées (Zhao *et al.*, 2007; Hassan *et al.*, 2007; Giuliano *et al.*, 2007; Yuret, 2007; Dahl *et al.*, 2007; Mohammad *et al.*, 2007; Hawker, 2007) ou semi-supervisées (Martinez *et al.*, 2007; Hassan *et al.*, 2007) et, (b) d'autre part, les méthodes entièrement non-supervisées qui ne reposent que sur l'analyse de corpus de textes sans ressources lexicales pour

1. http://www.irit.fr/semdis2014/fr/

prédéfinir l'inventaire des substituts possibles (Van de Cruys *et al.*, 2011). Le premier type d'approche est, de loin, le plus fréquent.

Dans cet article nous présentons une batterie de méthodes « simples » basées sur les balades aléatoires dans les réseaux lexicaux qui, par un système d'agrégation adapté à la tâche, constituent les méthodes proposées pour SemDis2014 (une supervisée [2] et une non-supervisée [3]). Notons toutefois que dans nos méthodes supervisées l'inventaire des candidats comprend la totalité des unités lexicales constitutives de la ressources lexicales utilisée (aucun filtre n'est utilisé).

La section 2 décrit le fonctionnement général de ces méthodes, la section 2.2 explique comment les balades aléatoires dans les réseaux génèrent une « vision » de proximité d'un sommet quelconque du réseau sur le reste de son réseau. Après avoir présenté en section 3 les réseaux lexicaux à partir desquels nous calculons ces « visions » de proximité, nous décrivons en section 4 l'ensemble des méthodes simples et leurs agrégations en deux méthodes supervisée et non-supervisée soumises à la tâche 1 de SemDis2014 ainsi que les résultats de leurs évaluations sur cette tâche. Enfin, en section 5, nous comparons, avec des données simples et controlées, les visions par balades aléatoires courtes aux visions par similarités construites sur une analyse en composantes principales. Nous concluons en section 6.

2 Méthodologie

Les neuf méthodes BACANAL exposées dans ce papier sont toutes bâties à partir de balades aléatoires courtes dans des réseaux unipartis ou bipartis construits sur diverses ressources. Si au moins un des graphes utilisés pour une méthode a été construit à partir de ressources produites « à la main » par des lexicographes ou par les foules alors cette méthode sera dite *supervisée*, et *non supervisée* [4] si elle n'utilise que des graphes construits automatiquement à partir de corpus.

Pour une phrase P dont ω est le mot à substituer, une méthode *simple* M_i sur un réseau lexical G produit un vecteur de réels $M_i(G,P,\omega)$ sur un ensemble V de mots de même partie du discours (PoS) que ω (ces méthodes sont dites *simples* dans la mesure où elles n'utilisent qu'un seul réseau lexical).

Deux méthodes M_i et M_j peuvent être *agrégées* en une méthode M_k. Par exemple si M_i est une méthode simple appliquée sur un réseau lexical G_1 et M_j est une méthode simple appliquée sur un réseau lexical G_2 alors l'agrégation des deux méthodes M_i et M_j consiste à combiner les deux vecteurs $M_i(G_1,P,\omega)$ et $M_j(G_2,P,\omega)$ en un vecteur $Agreg(M_i(G_1,P,\omega),M_j(G_2,P,\omega))$ qui, comme les deux vecteurs $M_i(G_1,P,\omega)$ et $M_j(G_2,P,\omega)$, est un vecteur de réels sur le même ensemble V de mots de même PoS que ω. Différents types d'agrégations peuvent être utilisées et sont décrites en section 4.2.

Par exemple, pour le mot ω =*fonde* à remplacer dans la phrase [5] P =« *Et cette confiance fonde la responsabilité du praticien.* », la méthode ϑ_9 exposée en section 4.2, fournit un vecteur dont les 10 verbes de plus fortes coordonées rangés en ordre décroissant sont :

établir, **constituer**, **créer**, former, instituer, **assurer**, mettre, **instaurer**, **poser**, construire.

La méthode ϑ_9 est celle qui, selon les évaluations de SemDis2014, propose les meilleures listes de substituts (les mots en gras sont les subtituts de *fonde* dans la phrase P qui ont été proposés par la méthode ϑ_9 et par au moins deux des évaluateurs de SemDis2014).

Nous exposons ci-dessous le cœur des méthodes BACANAL bâties à partir de balades aléatoires courtes dans des réseaux lexicaux.

2.1 Notations préliminaires

Un graphe $G=(V,E)$ est la donnée d'un ensemble non vide fini V de sommets, et d'un ensemble $E \subseteq V \times V$ de couples de sommets formant des arêtes :
– $n=|V|$ est l'*ordre* de G (son nombre de sommets),

2. Méthode qui s'appuie sur des ressources lexicales de type dictionnairique.
3. Méthode de catégorie (b).
4. Cette dénomination peut cependant être abusive dans la mesure où le système automatique pourrait éventuellement utiliser dans sa chaîne de traitements des ressources construites « à la main », par exemple quand la chaîne de traitements utilise un analyseur syntaxique qui lui même utilise des ressources construites « à la main ».
5. C'est la phrase numéro 93 du jeu de test fourni par SemDis2014.

– $m = |E|$ est la *taille* de G (son nombre d'arêtes),
– le graphe est *biparti* lorsqu'il existe deux ensembles $V_\top \subset V$ et $V_\bot \subset V$ tels que :
  – $V_\top \cup V_\bot = V$ et $V_\top \cap V_\bot = \emptyset$: V est l'union de deux ensembles d'intersection vide ;
  – $E \subseteq (V_\top \times V_\bot) \cup (V_\bot \times V_\top)$: il n'existe pas d'arête entre les sommets de $V_\bot$ ni entre les sommets de $V_\top$.
  On notera alors un tel graphe biparti : $G = (V_\top, V_\bot, E)$. Par ailleurs, un graphe $G = (V,E)$ est dit *pondéré* lorsque chaque arête $(r,s) \in E$ est valuée par un poids $w(r,s) \in \mathbb{R}^+$. On notera alors un tel graphe pondéré $G = (V,E,w)$.

2.2 Balades aléatoires

Soit $G = (V,E,w)$ un graphe pondéré de n sommets et m arêtes où chaque arête $(i,j) \in E$ est pondérée par un poids $w(i,j) \in \mathbb{R}^+$. On attribue à chaque sommet du graphe un vecteur de coordonnées dans $\mathbb{R}^n$ qui représente la « vision » qu'a le sommet en question sur le reste du graphe. Pour modéliser la « vision » qu'a un sommet sur le reste du graphe, nous considérons un marcheur se baladant aléatoirement en suivant les arêtes du graphe. La distribution de probabilité de la position de ce marcheur est donnée par la chaîne de Markov associée au graphe. Cette chaîne de Markov est définie par la matrice de transition A (équation 1) où $W(u)$ est la somme des poids des arêtes partant de u, soit $W(u) = \sum_{v \in V} w(u,v)$.

$$A = (a_{u,v})_{u,v \in V} \quad avec \quad a_{u,v} = \begin{cases} \frac{w(u,v)}{W(u)} & \text{si } (u,v) \in E \\ 0 & \text{sinon} \end{cases} \tag{1}$$

Si P_0 est la distribution de probabilité initiale du marcheur (c'est-à-dire un vecteur de dimension $n = |V|$ où $[P_0]_u$ est la probabilité de présence sur u au temps $t = 0$) alors la distribution de probabilité du marcheur après t pas est $P_t = P_0 A^t$ (le produit du vecteur P_0 de dimension n par la matrice A^t de dimension $n \times n$.)

Pour modéliser la « vision » qu'a un sommet $u \in V$ à un instant t donné sur le reste du graphe G, on définit le vecteur $\vartheta(G,u,t) = \delta_{\{u\}} A^t$ comme la distribution de probabilité d'un marcheur ayant effectué t pas depuis u, où δ_X est l'équiprobabilité d'être sur un des sommets de X (δ_X est un vecteur-ligne de dimension $|V|$ contenant la valeur 0 sur toutes ses coordonnées, excepté celles correspondant aux sommets de X qui valent $\frac{1}{|X|}$).

Si $[\vartheta(G,u,t)]_r > [\vartheta(G,u,t)]_s$ c'est que le sommet u « voit mieux » le sommet r que le sommet s, et la « vision » qu'a le sommet u en question sur les sommets de V est entièrement gouvernée par la structure du graphe $G = (V,E,w)$.

Si le graphe est apériodique, ce vecteur $\vartheta(G,u,t)$ converge quand $t \to \infty$. Cette limite correspond en fait à la version la plus simple du PageRank (Brin & Page, 1998; Manning *et al.*, 2008). Notons que cette limite ne dépend plus du sommet de départ, c'est-à-dire que $\forall u,r \in V, \lim_{t\to\infty} \vartheta(G,u,t) = \lim_{t\to\infty} \vartheta(G,r,t)$ et donne une information globale [6] sur le graphe (quels sont les sommets les plus « importants »).

À l'inverse, pour $t = 1$, $\vartheta(G,u,1)$ correspond à une version normalisée du vecteur d'adjacence du sommet u. Cette information est alors complètement locale, puisque ce vecteur ne dépend que du strict voisinage de u (u ne voit que ses voisins). Il est possible d'utiliser ce vecteur comme modèle, on a alors une modélisation vectorielle classique. Cependant cette modélisation ne prend en compte qu'une vision extrêmement locale de la topologie du graphe depuis u.

En revanche, lorsqu'on effectue des balades de temps courts ($3 \leqslant t \leqslant 8$), $\vartheta(G,u,t)$ dépend d'un voisinage plus large. Dans ce cas, même si deux sommets n'ont aucun voisin immédiat en commun, la ressemblance potentielle des voisins de leurs voisins peut amener ces deux sommets à « mieux se reconnaître ». $\vartheta(G,u,t)$ est alors une « vision de proximité », un compromis, entre une « vision trop locale » ($t = 1$) et une « vision trop globale » ($t \to \infty$).

Afin de généraliser la « vision » que peut avoir un ensemble S quelconque à un instant t donné sur le reste du graphe G, on définit le vecteur :

$$\vartheta(G,S,t) = \begin{cases} \delta_{S\cap V} A^t & \text{si } S \cap V \neq \varnothing \\ \vec{0} & \text{sinon } \textit{où } \vec{0} \textit{ est le vecteur nul de dimension } |V| \end{cases} \tag{2}$$

6. Tout sommet a alors la même « vision ». Par exemple si G est un graphe non pondéré, réflexif et symétrique, alors le sommet qui est toujours « le mieux vu » par tous les autres sommets est le sommet de plus fort degré (voir (Gaume, 2004)).

3 Réseaux lexicaux

Deux types de réseaux lexicaux ont été utilisés pour la construction de nos méthodes BACANAL : (a) des réseaux construits à partir ressources de type dictionnairiques réalisées à la main par des lexicographes ou par les foules et (b) des ressources construites par analyse distributionnelle de corpus de textes.

Réseaux lexicaux construits à partir de la ressource *DicoSyn* : La ressource *DicoSyn* a été construite lors d'un projet collaboratif entre IBM et l'Institut National de la Langue Française [7]. A partir de sept dictionnaires classiques (Bailly, Benac, Du Chazaud, Guizot, Lafaye, Larousse et Robert) ont été extraites les relations synonymiques, puis le graphe ainsi obtenu **Gdsyn** a été réflexivisé, symétrisé et catégorisé par PoS en trois graphes $\mathbf{Gdsyn_A}$, $\mathbf{Gdsyn_N}$, $\mathbf{Gdsyn_V}$, les caractéristiques générales de ces graphes sont décrites dans la table 2.

Réseaux lexicaux construits à partir de la ressource *Jeux De Mots* : La ressource *Jeux De Mots* [8] est construite par les foules en utilisant un jeu décrit dans (Lafourcade, 2007). Les joueurs doivent trouver le plus de mots possible qui sont associés à un terme présenté à l'écran, selon une règle prévue par le jeu. Le but est de trouver autant d'associations sémantiques que possible que les autres joueurs ont trouvées, mais que le joueur concurrent n'a pas trouvées. Plusieurs règles peuvent être proposées, y compris la *synonymie* et l'*association libre*. Les résultats recueillis en janvier 2014 permettent de construire un graphe de mots liés par des relations sémantiques typées (selon les règles du jeu). $\mathbf{GjdmS_A}$, $\mathbf{GjdmS_N}$, $\mathbf{GjdmS_V}$ sont les graphes de synonymie et **GjdmA** est le graphe d'association libre, tous les quatre construits à partir de la ressource *Jeux De Mots*. Ces quatre graphes sont réflexivisés, symétrisés et non pondérés et leurs caractéristiques générales sont décrites dans la table 2.

Réseaux lexicaux construits à partir de la ressource *LM10* : La ressource *LM10* construite par Benoît Habert est un corpus de 200 millions de mots, constitué des articles du journal *Le Monde* des années 1991 à 2000.

Une analyse syntaxique en dépendance de LM10 a été réalisée au sein du laboratoire CLLE [9] par l'analyseur syntaxique probabiliste *Talismane* [10] (Urieli, 2013). Pour fonctionner dans une langue **L** donnée, *Talismane* a besoin d'un lexique de **L** [11], d'un ensemble d'étiquettes des parties du discours de **L** [12] , d'un ensemble de traits et d'un ensemble de règles spécifiques à **L**. La version que nous utilisons ici a été entraînée pour le français sur le French TreeBank [13] (Abeillé *et al.*, 2003). En entrée, *Talismane* prend un texte brut et, en sortie, il produit une liste de tokens : identifiant du token (id), lemme, forme, PoS, caractéristiques grammaticales (CG), identifiant du recteur du token (GOV), nature de la relation de dépendance $(token, recteur)$ (REL). Par exemple, l'analyse par *Talismane* de l'énoncé *« Et cette confiance fonde la responsabilité du praticien. »* produit la sortie décrite dans le tableau 1. *Talismane* fait une analyse en dépendance de

ID	FORME	LEMME	POS	CG	GOV	REL
1	Et	et	CC	_	0	root
2	cette	cette	DET	g=f|n=s	3	det
3	confiance	confiance	NC	g=f|n=s	4	suj
4	fonde	fonder	V	n=s|p=13|t=pst	1	dep_coord
5	la	la	DET	g=f|n=s	6	det
6	responsabilité	responsabilité	NC	g=f|n=s	4	obj
7	du	de	P+D	g=m|n=s	6	dep
8	praticien	praticien	NC	g=m|n=s	7	prep
9	.	.	PONCT	_	1	ponct

TABLE 1 – Sorties de *Talismane* pour la phrase : *« Et cette confiance fonde la responsabilité du praticien. »*

7. Aujourd'hui ATILF :http://www.atilf.fr/
8. http://www.lirmm.fr/jeuxdemots/jdm-accueil.php
9. http://w3.erss.univ-tlse2.fr/
10. http://redac.univ-tlse2.fr/applications/talismane.html
11. Le Lefff (Sagot *et al.*, 2006) pour la version utilisée ici.
12. Étiquettes en grande partie reprises de (Crabbé & Candito, 2008) pour la version utilisée ici.
13. http://www.llf.cnrs.fr/Gens/Abeille/French-Treebank-fr.php

surface entre tous les tokens mis en jeu, ponctuation comprise, et chaque token ne peut avoir qu'un seul recteur. Afin d'identifier toutes les relations syntaxiques logiques entre tokens, les sorties de *Talismane* sont passées à un module de déduction qui calcule :

- la relation de coordination entre tokens coordonnés :
  une pomme et une poire $\rightarrow \langle NC.pomme, coor\_dep, NC.poire \rangle$
- la relation entre un token et tous ses dépendants lorsque ceux-ci sont coordonnés :
  il joue et chante $\rightarrow \langle V.jouer, suj, PRO.il \rangle, \langle V.chanter, suj, PRO.il \rangle$
- la relation entre un token et tous ces gouverneurs lorsque ceux-ci sont coordonnés :
  il mange une pomme et une poire $\rightarrow \langle V.manger, obj, NC.pomme \rangle, \langle V.manger, obj, NC.poire \rangle$
- la relation *suj* (resp. *obj*) entre le sujet (resp. objet) logique et le verbe lorsque le sujet (resp. objet) réel est un pronom relatif :
  le gars qui joue au foot $\rightarrow \langle V.jouer, suj, NC.gars \rangle$
- la relation *Prep* [14] entre un nom ou un verbe et la tête du syntagme prépositionnel qui le complète lorsqu'un syntagme prépositionnel complète un nom ou un verbe :
  c'est un train à vapeur $\rightarrow \langle NC.train, Prep/\text{à}, NC.vapeur \rangle$
- la relation *mod* entre les noms et leurs attributs du sujet :
  le livre est rouge $\rightarrow \langle NC.livre, mod, ADJ.rouge \rangle$
- la relation *obj* entre le complément d'objet logique d'un verbe et ce verbe lorsque le verbe est à la forme passive : *la souris est mangée par le chat* $\rightarrow \langle V.manger, obj, NC.souris \rangle$
- la relation *suj* entre les participes présents et leur sujet :
  l'avocat plaidant une cause $\rightarrow \langle V.plaider, suj, NC.avocat \rangle$

Ce module de déduction pronominalise aussi les verbes qui ont un complément d'objet clitique troisième personne et réétiquette certaines parties du discours : les verbes étiquetés *verbe infinitif*, *verbe impératif*, *verbe subjonctif* et *participe présent* par *Talismane* sont simplement réétiquetés *verbe*.

Trois graphes $Glm10_N$, $Glm10_A$, $Glm10_V$ (c.f. tableau 3) sont ensuite construits à partir des sorties de *Talismane* enrichies par le module de déduction comme suit. Définissons :

- C_l l'ensemble des contextes syntaxiques d'un lemme l dans LM10 : $C_l = \{(rel, l_c)\}$ tels que l_c est syntaxiquement lié à l par *rel* dans LM10 ;
- C l'ensemble des contextes syntaxiques de LM10 : $C = \bigcap_{l \in L} C_l$.

Soit $PoS \in \{A, N, V\}$, $Glm10_{PoS} = (L_{PoS} \cup C, E)$ est un graphe biparti tel que $\{l, c\} \in E \Leftrightarrow c \in C_l$. Toute arête $\{l, c\} \in E$ est pondérée par une mesure de type information mutuelle *IM* entre le lemme l et le contexte c :

$$IM = \frac{freq((*,*)) \times freq((l,c))}{freq((l,*)) \times freq((*,c))} \tag{3}$$

Réseaux lexicaux construits à partir de la ressource *frWaC* : La ressource *frWaC* [15] qui est décrite dans (Baroni *et al.*, 2009) est un corpus de 1.6 milliard de mots construit à partir du Web en limitant l'analyse au domaine .fr. Les graphes **$Gfrwac_A$ $Gfrwac_N$ $Gfrwac_V$** (c.f. tableau 3) ont été construits de la même manière que les graphes **Glm10**.

	$Gdsyn_A$	**$Gdsyn_N$**	**$Gdsyn_V$**	**$GjdmS_A$**	**$GjdmS_N$**	**$GjdmS_V$**	**GjdmA**
n	9 452	29 372	9 147	9 859	29 213	7 658	153 586
m	42 403	100 759	51 423	30 088	56 383	22 262	928 399

TABLE 2 – Caractéristiques des graphes unipartis : n est le nombre de sommets, m le nombre total d'arêtes.

Nous indiquons ci-dessous le voisinage « immédiat » du verbe *fonder* dans les graphes présentés ci-dessus :

Dans $Gdsyn_V$, *fonder* a 32 **voisins :** affermir, appuyer, asseoir, assurer, baser, bâtir, commencer, compter, constituer, construire, créer, engendrer, enter, forger, former, instaurer, instituer, justifier, lancer, mettre, motiver, organiser, ouvrir, placer, poser, reposer, tabler, échafauder, édifier, élever, ériger, établir

14. Il y a autant de relation *Prep* que de prépositions.
15. http://wacky.sslmit.unibo.it/doku.php?id=corpora

	$\mathbf{Glm10_A}$	$\mathbf{Glm10_N}$	$\mathbf{Glm10_V}$	$\mathbf{Gfrwac_A}$	$\mathbf{Gfrwac_N}$	$\mathbf{Gfrwac_V}$
n	57 623	520 355	223 843	134 559	964 769	319 249
$\mathbf{n}_l$	21 181	48 491	8 017	55 771	133 506	18 734
$\mathbf{n}_c$	36 442	471 864	215 826	78 788	831 263	300 515
m	872 464	7 556 008	2 654 104	958 138	8 643 588	2 151 146

TABLE 3 – Caractéristiques des graphes bipartis : n est le nombre total de sommets, n_l le nombre de lemmes, n_c le nombre de contextes, m le nombre d'arêtes.

Dans $\mathbf{GjdmS_V}$, *fonder* a 15 **voisins :** affermir, appuyer, asseoir, assoir, baser, bâtir, constituer, créer, former, instaurer, instituer, justifier, édifier, élever, ériger

Dans GjdmA, *fonder* a 47 **voisins :** acte fondateur, amorcer, aménager, assurer, attaquer, commencer, composer, concevoir, construction, construire, disposer, débuter, démarrer, enfanter, engendrer, engrener, entamer, entreprendre, entreprise, esquisser, fixer, fondateur, fondation, fondement, foyer, imaginer, implanter, installer, inventer, maison, mettre, montrer, partir, placer, poser, presser, production, produire, préluder, réaliser, se fonder, ébaucher, échafauder, élaborer, équilibrer, établir, étrenner

Dans $\mathbf{Glm10_V}$, *fonder* a 583 **voisins** [16] **:** NC.espoir.Dep.obj (freq=144, IM=120.347), NC.société.Dep.obj (freq=138, IM=59.6531), NC.famille.Dep.obj (freq=98, IM=77.4457), NC.revue.Dep.obj (freq=85, IM=317.996), V.venir.Gov.Prep/de (freq=82, IM=7.29996), NC.principe.Dep.suj (freq=82, IM=84.03), NC.compagnie.Dep.obj (freq=79, IM=120.454), NC.parti.Dep.obj (freq=78, IM=46.1816), NC.association.Dep.obj (freq=78, IM=100.757), NC.valeur.Dep.suj (freq=76, IM=73.5628)

Dans $\mathbf{Gfrwac_V}$, *fonder* a 824 **voisins** [17] **:** NC.famille.Dep.obj (freq=1144, IM=387.762), NC.monastère.Dep.obj (freq=517, IM=4889.38), NC.société.Dep.obj (freq=472, IM=137.059), NC.groupe.Dep.obj (freq=363, IM=71.3103), NC.école.Dep.obj (freq=283, IM=126.836), NC.action.Dep.obj (freq=270, IM=26.9095), V.être.Gov.Prep/de (freq=264, IM=1.91398), V.permettre.Gov.Prep/de (freq=253, IM=2.75423), NC.association.Dep.obj (freq=202, IM=77.0306), NC.compagnie.Dep.obj (freq=200, IM=383.908)

4 Méthodes

Dans une phrase P soit ω un mot cible de P et C_P^ω l'ensemble des contextes syntaxiques de ω dans P identifié par *Talismane* + module de déduction. Par exemple, soit $P =$*« Et cette confiance fonde la responsabilité du praticien. »* et $\omega =$*fonde* le mot-cible de P, le mot *fonde* a trois contextes syntaxiques dans cette phrase [18] (voir tableau 1) :
$C_P^\omega = \{(NC.confiance, Dep.suj), (NC.responsabilité, Dep.obj), (CC.et, Gov.dep\_coord)\}$

4.1 Visions simples

Nous présentons dans le tableau 4 sept méthodes qui utilisent des visions simples sur différents graphes lexicaux. Chaque méthode construit une liste ordonnée de lemmes d'un des trois types suivants :
- $\mathbf{T_1}$, liste ordonnée sur l'axe paradigmatique de ω par rapport à ω et indépendamment du contexte C_P^ω de la phrase P (c'est à dire couvrant potentiellement l'ensemble de la polysémie du mot ω) ;
- $\mathbf{T_2}$, liste ordonnée sur l'axe syntagmatique de C_P^ω par rapport C_P^ω et indépendamment de ω ;
- $\mathbf{T_3}$, liste ordonnée par rapport à ω sur axe non typé (les relations entre deux lemmes peuvent être paradigmatiques ou syntagmatiques) et indépendamment du contexte C_P^ω de la phrase P.

Les méthodes ϑ_1, ϑ_2 et ϑ_3 sont supervisées tandis que les méthodes ϑ_4, ϑ_5, ϑ_6 et ϑ_7 sont non-supervisées

16. Ces voisins sont les contextes de *fonder* dans *LM*10, nous présentons ici les 10 plus fréquents, ('freq' est la fréquence du contexte avec *fonder*, et 'IM' est le poids de l'arête entre *fonder* et le contexte).
17. Ces voisins sont les contextes de *fonder* dans $frWaC$, nous présentons ici les 10 plus fréquents.
18. Le module de dépendance ne change pas les sorties de *Talismane* pour cette phrase.

Méthode	Type de liste
$\boldsymbol{\vartheta}_1 = \vartheta(Gdsyn,\ \{\omega\},\ 3)$	$\mathbf{T_1}$
$\boldsymbol{\vartheta}_2 = \vartheta(GjdmS,\ \{\omega\},\ 3)$	$\mathbf{T_1}$
$\boldsymbol{\vartheta}_3 = \vartheta(GjdmA,\ \{\omega\},\ 3)$	$\mathbf{T_3}$
$\boldsymbol{\vartheta}_4 = \vartheta(Glm10,\ \{\omega\},\ 2)$	$\mathbf{T_1}$
$\boldsymbol{\vartheta}_5 = \vartheta(Gfrwac,\ \{\omega\},\ 2)$	$\mathbf{T_1}$
$\boldsymbol{\vartheta}_6 = \vartheta(Glm10,\ C_P^{\omega},\ 3)$	$\mathbf{T_2}$
$\boldsymbol{\vartheta}_7 = \vartheta(Gfrwac,\ C_P^{\omega},\ 3)$	$\mathbf{T_2}$

TABLE 4 – Sept visions simples

4.2 Agrégations de visions simples

Le but de la tâche 1 de SemDis2014 est la substitution lexicale : *remplacer un mot ω dans une phrase P par un autre mot tout en préservant au maximum le sens de la phrase P*. Aucune des sept visions simples décrites ci-dessus ne peut espérer remplir cette tâche avec succès, ce n'est d'ailleurs pas leurs buts.

On peut cependant espérer s'approcher au mieux de la tâche de substitution lexicale en combinant plusieurs visions simples. Par exemple en multipliant coordonnées par coordonnées les deux vecteurs issus de deux méthodes de type T_1 et T_2 on peut espérer renforcer l'axe paradigmatique du mot ω sur le sens qu'il prend dans le contexte C_P^{ω} de la phrase P.

Pour aller dans ce sens nous définissons ci-dessous trois façons de combiner les méthodes [19]. Soit A et B deux vecteurs de même dimension :

Agreg$_1$ $(\mathbf{A},\mathbf{B})$:

$$Agreg_1(A,B) = [C]_i = \begin{cases} [A]_i \,.\, [B]_i & \text{si } [A]_i \neq 0 \text{ et } [B]_i \neq 0 \\ [A]_i & \text{sinon} \end{cases} \tag{4}$$

Agreg$_2$ $(\mathbf{A},\mathbf{B})$:

$$Agreg_2(A,B) = [C]_i = \begin{cases} [B]_i & \text{si } [A]_i = 0 \\ [A]_i & \text{sinon} \end{cases} \tag{5}$$

Agreg$_3$ $(\mathbf{A},\mathbf{B})$:

$$Agreg_3(A,B) = [C]_i = \begin{cases} [B]_i & \text{si } [A]_i \neq 0 \\ 0 & \text{sinon} \end{cases} \tag{6}$$

Nous pouvons maintenant définir les deux méthodes que nous avons soumises à SemDis2014 :

Méthode non supervisée : $\boldsymbol{\vartheta}_8 = Agreg_1(Agreg_1(\vartheta_4,\ \vartheta_5),\ \ Agreg_1(\vartheta_6,\ \vartheta_7))$
Méthode supervisée : $\boldsymbol{\vartheta}_9 = Agreg_2(Agreg_1(Agreg_1(\vartheta_2,\vartheta_3),\vartheta_6),\ \ Agreg_3(Agreg_2(\vartheta_1,\vartheta_2),\ Agreg_1(Agreg_1(\vartheta_2,\vartheta_3),\vartheta_6)))$

Le tableau 5 résume les résultats des méthodes exposées ici sur la phrase numéro 93.

4.3 Évaluation

Parmis les 10 méthodes soumises par l'ensemble des participants à Semdis14, la méthode $\boldsymbol{\vartheta}_9$ est celle qui, selon les évaluations de SemDis2014 sur un ensemble de 300 phrases avec 30 cibles à désambiguïser (10 verbes, 10 noms, 10 adjectifs avec 10 phrases par cible), propose les meilleures listes de substituts. Les résultats des méthodes ont été évalués à l'aide de deux mesures de rappel : *best* et *oot* définies par (McCarthy & Navigli, 2009) : soit H l'ensemble des annotateurs SemDis2014, T l'ensemble des phrases avec au moins deux substituts proposés par les annotateurs, h_i l'ensemble des réponses produites par les annotateurs pour une phrase $i \in T$, A l'ensemble des phrases de T pour lesquels le système produit au moins une réponse, a_i l'ensemble des substituts proposés par le système pour une phrase $i \in T$, H_i l'union

19. Il se peut qu'une méthode M_1 donne en générale de meilleurs resultats qu'une autre methode M_2, mais que la méthode M_1 ai une moins bonne couverture lexicale que la méthode M_2. C'est principalement pour cette raison que les méthodes d'agrégations ne sont pas symétriques.

Gold	**créer, forger, constituer, justifier, être à la base, entraîner, assurer, impliquer, baser, instaurer, induire, définir, être à l'origine, établir, installer, poser, supporter**
ϑ_1	**établir**, bâtir, **créer**, construire, faire, organiser, former, **constituer**, élever, placer
ϑ_2	bâtir, **constituer**, **créer**, élever, **établir**, édifier, ériger, **instaurer**, instituer, appuyer
ϑ_3	responsabilité, confiance, charge, meilleur ami, sureté, affect, condamnation, devoir, poids, dette
ϑ_4	fondre, diriger, présider, animer, **créer**, rejoindre, perpétuer, abriter, érier, racheter
ϑ_5	se marier, ème, rejoindre, pondre, échanger, arranger, engendrer, dater, rivaliser, diriger
ϑ_6	se décréter, se mériter, se rétablir, endosser, se rejeter, se démentir, se renvoyer, saisissant, se évanouir, imputer
ϑ_7	généraliser, se mériter, se acquérir, aveugler, se rejeter, endosser, décliner, régner, se décréter, assumer
ϑ_8	reposer, se installer, se fonder, assumer, diriger, régner, rejoindre, quitter, animer, se appuyer
ϑ_9	**établir**, **constituer**, **créer**, former, instituer, **assurer**, mettre, **instaurer**, **poser**, construire

TABLE 5 – Résultats sur la phrase numéro 93 : *« Et cette confiance <fonde> la responsabilité du praticien. »*

des h_i et freq(s) le nombre d'occurrences du substitut $s \in H_i$ dans H_i. Une première mesure *best* définie par l'équation 7 indique le rappel au rang 1 de la méthode par rapport à des solutions de référence proposées par les organisateurs de SemDis2014. La seconde mesure *oot* (pour *out of best*) définie par l'équation 7 indique le rappel au rang 10 de la méthode sans prendre en compte l'orde des réponses. Les résultats obtenus par les méthodes présentées dans ce papier sur la phrase numéro 93 sont détaillés dans le tableau 5.

$$best = \frac{\sum_{a_i : i \in T} \frac{\sum_{s \in a_i} freq(s)}{|a_i|.|H_i|}}{|T|} \quad oot = \frac{\sum_{a_i : i \in T} \frac{\sum_{s \in a_i} freq(s)}{|H_i|}}{|T|} \tag{7}$$

Le tableau 6 présente les résultats des méthodes simples ainsi que des méthodes ϑ_8 (non-supervisée) et ϑ_9 (supervisée) soumises à SemDis2014 :

Méthodes	**Type**	**best**	**oot**
$\vartheta_1 = \vartheta(Gdsyn,\ \{\omega\},\ 3)$	supervisée	.0453	.3245
$\vartheta_2 = \vartheta(GjdmS,\ \{\omega\},\ 3)$	supervisée	.0645	.3519
$\vartheta_3 = \vartheta(GjdmA,\ \{\omega\},\ 3)$	supervisée	.0022	.0736
$\vartheta_4 = \vartheta(Glm10,\ \{\omega\},\ 2)$	non-supervisée	.0259	.1347
$\vartheta_5 = \vartheta(Gfrwac,\ \{\omega\},\ 2)$	non-supervisée	.0319	.0799
$\vartheta_6 = \vartheta(Glm10,\ C_P^{\omega},\ 3)$	non-supervisée	.0061	.0368
$\vartheta_7 = \vartheta(Gfrwac,\ C_P^{\omega},\ 3)$	non-supervisée	.0024	.0228
ϑ_8	non-supervisée	.0511	.2129
ϑ_9	supervisée	.0970	.4017

TABLE 6 – Résultats des méthodes BACANAL

Le tableau 6 met en évidence une amélioration significative par les méthodes agrégées des performances des méthodes simples sur lesquelles elles reposent :
– ϑ_8 / ϑ_5 : best : +60% ; oot : +166%
– ϑ_9 / ϑ_2 : best : +50% ; oot : +14%
Notons toutefois que les deux méthodes basées sur des ressources paradigmatiques construites à la main [20] (ϑ_1 basée sur

20. Nous ne considérons pas l'association libre comme une relation paradigmatique puisqu'elle met en jeu des relations entre parties du discours distinctes.

Gdsyn et ϑ_2 basée sur GjdmS), sont performantes au rang 10 (environ 32% des réponses fournies par au moins deux annotateurs sont trouvées par ϑ_1 et 35% par ϑ_2) et que la méthode ϑ_9 n'améliore les performances de ϑ_2 que de 14% au rang 10. Toutefois, au rang 1, ϑ_9 est significativement plus performante que ϑ_2.

5 Vers une comparaison avec les méthodes opérant par réduction de dimension

Toutes les méthodes exposées ici sont bâties à partir de balades aléatoires courtes dans des réseaux unipartis ou bipartis construits sur diverses ressources. Cependant, un réseau lexical uniparti $G = (V, E, w)$ peut être vu comme une matrice lexicale de dimension $|V| \times |V|$: $M_G = (a_{u,v})_{u,v \in V}$, avec $a_{u,v} = w(u,v)$, et un réseau lexicale biparti $G = (V_\top, V_\bot, E, w)$ peut être vu comme une matrice lexicale de dimension $|V_\top| \times |V_\bot|$: $M_G = (a_{u,v})_{u \in V_\top, v \in V_\bot}$, avec $a_{u,v} = w(u,v)$.

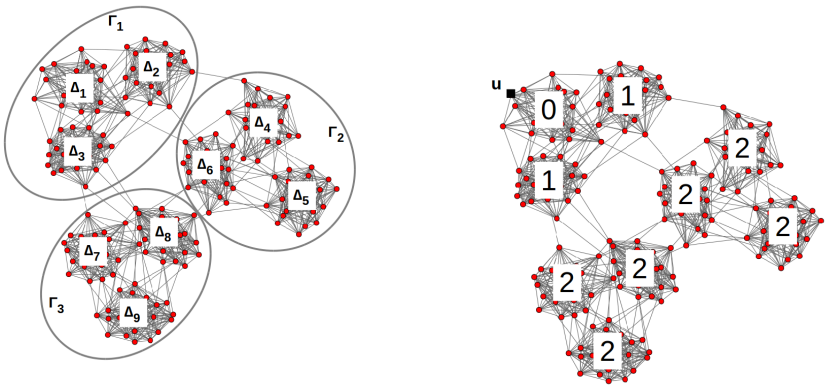
Beaucoup de méthodes (Van de Cruys *et al.*, 2011; Erk & Padó, 2009, 2010; Dinu & Lapata, 2010; Thater *et al.*, 2010) commencent par réduire la matrice M_G en une matrice M_G^k de dimensions k avec des méthodes d'analyse en composantes principales (ACP) puis calculent une similarité entre les vecteurs de M_G^k, par exemple : $cos([M_G^k]_u, M_G^k]_v)$. C'est alors le vecteur $\varphi(G,u,k) = (a_v)_{v \in V}$, avec $a_v = cos([M_G^k]_u, M_G^k]_v)$ qui est utilisé comme « vision » de u.

Nous proposons ci-dessous de comparer, sur un graphe artificiel simple et controlé, les méthodes BACANAL à celles qui utilisent une réduction d'espaces vectoriels. Une telle comparaison ne remplace en aucun cas une comparaison sur des données réelles. Cependant sur ces données controlées un résultat précis est attendu. Cela permet donc de comparer les résultats de chaque méthode par rapport aux résultats attendus. Ceci est un premier pas pour mieux comprendre les ressemblances et différences existantes entre les méthodes.

Pour comparer les méthodes nous utilisons un modèle de graphe artificiel composé de deux niveaux de clusterisation : les sommets sont regroupés en trois gros clusters, eux-même décomposables en trois petits clusters. Formellement, nous utilisons un graphe $G = (V,E)$ tel que V est l'union de $k = 9$ ensembles $\Delta_1, \ldots \Delta_9$ de $n = 20$ sommets chacun [21]. Ces sommets sont regroupés en trois ensembles $\Gamma_1 = \Delta_1 \cup \Delta_2 \cup \Delta_3$, $\Gamma_2 = \Delta_4 \cup \Delta_5 \cup \Delta_6$, $\Gamma_3 = \Delta_7 \cup \Delta_8 \cup \Delta_9$. Une arête e entre deux sommets u et v est créée avec la probabilité :

- $p_1 = 0.5$ si les deux sommets appartiennent à un même ensemble Δ ($\exists i$ tel que $u, v \in \Delta_i$) ;
- $p_2 = 0.01$ s'ils appartiennent à des ensembles Δ distincts mais à un même ensemble Γ ($\nexists i$ tel que $u, v \in \Delta_i$ mais $\exists j$ tel que $u, v \in \Gamma_j$) ;
- $p_3 = 0.001$ s'ils appartiennent à deux ensembles Γ distincts ($\nexists i$ tel que $u, v \in \Gamma_i$).

Un tel graphe est représenté dans la figure 1(a).



(a) $G = (V,E)$ (b) Etiquetage des sommets

FIGURE 1 – Graphe artificiel avec 3 zones denses Γ_1, Γ_2, Γ_3 où chacune de ces zones denses est constituée de 3 zones locales encore plus denses Γ_1 : Δ_1, Δ_2, Δ_3 ; Γ_2 : Δ_4, Δ_5, Δ_6 ; Γ_3 : Δ_7, Δ_8, Δ_9.

Notre idée est de comparer, pour un sommet u quelconque de G, les visions du graphe obtenues par chacune des méthodes

21. Si $i \neq j$ alors $\Delta_i \cap \Delta_j = \varnothing$.

à partir de u, à l'aide du critère suivant : l'ordre des sommets de G obtenus par ces méthodes (à partir de u) respecte-t-il l'ordre des clusters autour de u (Δ_u, Γ_u, G). En effet, on souhaite que les sommets du petit cluster Δ_u arrivent avant ceux du cluster moyen Γ_u et eux-mêmes avant les autres sommets du graphe.

Pour cela, nous allons observer les résulats de chaque métodes en notant *(0)* les sommets qui appartiennent à Δ_u ; *(1)* les sommets qui appartiennent à Γ_u mais pas à Δ_u ; *(2)* les sommets qui n'appartiennent pas à Γ_u (cet étiquetage est illustré dans la figure 1(b)). Ceci est une représentation « binaire » de ce qui peut se passer sur un graphe lexical dans une tâche de substitution.

Nous donnons ci-dessous les résultats pour une méthode utilisant des marches aléatoires courtes et différentes méthodes par réduction de dimension. La liste L_0 représente les étiquettes des sommets $v \in V$ quand elles sont ordonnées selon $\big(\vartheta(G,\ \{u\},\ 3)\big)_v$. Les listes L_3, L_9, L_{50} et L_{180} représentent les mêmes étiquettes mais ordonnées après une réduction de dimension (L_3 utilise $\big(\varphi(G,u,3)\big)_v$, L_9 utilise $\big(\varphi(G,u,9)\big)_v$, ...)

$\mathbf{L_0}$ =[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 2, 2, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]

$\mathbf{L_3}$ =[0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 1, 1, 1, 0, 1, 0, 1, 1, 1, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 1, 1,
1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]

$\mathbf{L_9}$ =[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2,
1, 1, 2, 1, 2, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]

$\mathbf{L_{50}}$ =[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 2, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1,
2, 2, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 2, 2, 2, 2, 2, 1, 2, 2, 2, 2, 2, 2, 1]

$\mathbf{L_{180}}$ =[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 2, 1, 1, 1,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 2, 2, 2, 2, 2, 2, 2, 1, 2, 2, 2, 2, 1, 2, 1, 2, 2]

Les listes L_0 et L_9 donnent toutes les deux, approximativement, les résultats « espérés » alors que ni la liste L_3 ni la liste L_{50} ni même la liste L_{180} (changement de base simple, matrice non réduite) ne sont satisfaisantes :

- La liste L_3 ne distingue pas suffisamment les sommets étiquetés 0 ($v \in \Delta_1$) des sommets étiquetés 1 ($v \notin \Delta_1$ mais $v \in \Gamma_1$). En réduisant la matrice à trois dimensions on a encore l'information de l'existence des trois gros agrégats Γ_1, Γ_2, Γ_3 mais celles des petits agrégats Δ_1, Δ_2 , Δ_3 est moins claire ;
- La liste L_{50} ne distingue pas suffisamment les sommets étiquetés 1 ($v \notin \Delta_1$ mais $v \in \Gamma_1$) des sommets étiquetés 2 ($v \notin \Gamma_1$). En réduisant la matrice à cinquante dimensions on a encore l'information de l'existence des trois petits agrégats Δ_1, Δ_2 , Δ_3, mais celles des trois gros agrégats Γ_1, Γ_2, Γ_3 est moins claire ;
- La liste L_{180} souffre des mêmes faiblesses que la liste L_{50}.

Nous voyons donc sur ce petit exemple que les méthodes par réduction de dimension sont semblables aux méthodes BACANAL à condition que le choix de k soit bien adapté à la ressource, ce qui est parfois difficile à réaliser sur des ressources réelles. Une étude comparative plus poussée des forces et faiblesses de ces méthodes sur des données lexicales réelles mériterait d'être systématisée. Notons qu'il est aussi possible de comprendre ces différences/ressemblances d'un point de vu algébrique. En effet, les marches aléatoires peuvent se comprendre comme un renforcement des valeurs propres les plus fortes [22], alors qu'une réduction de dimension ne conserve que les vecteurs qui ont les valeurs propres les plus fortes (Navarro, 2013).

6 Conclusion

Dans ce papier, nous avons présenté des méthodes bâties à partir de balades aléatoires courtes sur des réseaux lexicaux : les méthodes BACANAL. Selon les types de réseaux lexicaux qu'elles utilisent (construits à partir de corpus de textes ou à partir de ressources lexicales produites par des lexicographes ou par les foules), ces méthodes peuvent être supervisées ou entièrement non-supervisées. Un mécanisme d'agrégation de méthodes permet d'améliorer significativement les résultats

22. En effet, les marches aléatoires se basent sur la matrice A prise à la puissance t. Or A peut être décomposée sur une base de vecteurs propres : $A = UEV^T$ avec E la matrice diagonale des valeurs propres (on sait par ailleur que ces valeurs propres sont $<= 1$). On a donc que $A^t = UE^tV^T$ ce qui revient à renforcer les plus fortes valeurs propres. Une démonstration et une explication plus détailées restent cependant nécessaires.

des méthodes BACANAL simples sur lesquelles elles reposent (ex : ϑ_8 / ϑ_5 et ϑ_9 / ϑ_2).

Comme la plupart des méthodes de l'état de l'art, les évaluations *oot* de toutes les méthodes ayant concourues à la tâche 1 de Semdis14 sont inférieurs à 50% : ϑ_9 la meilleurs méthode selon les évaluations de Semdis14 obtient un *oot* égal à 0.4017. Obtenir un rappel élevé au rang 10 lors d'une tâche de substitution lexicale face à un gold construit à « à la main » semble donc être difficile.

Nous avons aussi amorcé une comparaison entre les méthodes par marche aléatoire courte et les méthodes par réduction de dimension et montré que les méthodes par réduction de dimension sont semblables aux méthodes BACANAL à condition que le choix de k soit bien adapté à la ressource. Un des avantages des méthodes BACANAL est que leur complexité est proportionnelle à la densité des graphes utilisés : une marche de temps t à partir d'un sommet quelconque d'un graphe de m arêtes se calcule avec une complexité $O(mt)$ (Navarro, 2013). Ainsi, la complexité des méthodes BACANAL sur des réseaux peu denses en arêtes telles que les réseaux lexicaux est faible. De plus, si les graphes sont trop larges, les méthodes de Monté-Carlo [23] peuvent facilement être utilisées pour calculer une approximation des marches aléatoires en temps court.

Toutefois, pour une tâche de substitution lexicale libre comme la tâche 1 de SemDis2014, la taille des graphes n'est pas le critère essentiel. En effet la qualité linguistique de ces graphes semble primer. Par exemple, les tableaux 2 et 3 montrent que les différences de résultats obtenus par les méthodes $\vartheta_4 = \vartheta(Glm10,\ \{\omega\},\ 2)$ ($best = .0259$ & $oot = .1347$) et $\vartheta_5 = \vartheta(Gfrwac,\ \{\omega\},\ 2)$ ($best = .0319$ & $oot = .0799$) ne sont pas liées à des différences de taille entre graphes utilisés, mais à des différences qualitatives entre les ressources sur lesquelles elles sont construites.

7 Remerciements

Nous remercions les organisateurs de SemDis2014 pour avoir proposé cette tâche et développé le matériel nécessaires aux évaluations. Nous remercions Franck Sajous et Assaf Urielli pour les nombreuses discussions toujours enrichissantes que nous avons eu ensemble et pour tous les pré-traitements sur les ressources que nous avons utilisées dans cet article (accessibles pour la plupart sur `http://redac.univ-tlse2.fr/`).

Références

ABEILLÉ A., CLÉMENT L. & TOUSSENEL F. (2003). Building a treebank for French. In A. ABEILLÉ, Ed., *Treebanks*, p. 165–188. Dordrecht : Kluwer.

BARONI M., BERNARDINI S., FERRARESI A. & ZANCHETTA E. (2009). The wacky wide web : a collection of very large linguistically processed web-crawled corpora. In *Proceedings of the Seventh International Language Resources and Evaluation (LREC'09)*, volume 43(3), p. 209–226.

BRIN S. & PAGE L. (1998). The anatomy of a large-scale hypertextual web search engine. *Computer Networks*, **30**(1-7), 107–117.

CRABBÉ B. & CANDITO M. (2008). Expériences d'analyses syntaxique statistique du français. In *Actes de la conférence TALN2008*, Avignon, France.

DAHL G., FRASSICA A. & WICENTOWSKI R. (2007). SW-AG : Local context matching for english lexical substitution. In *Proceedings of the 4th workshop on Semantic Evaluations (SemEval-2007)*, p. 304–307, Prague, Czech Republic.

DESALLE Y. (2012). *Réseaux lexicaux, métaphore, acquisition : une approche interdisciplinaire et inter-linguistique du lexique verbal.* PhD thesis, Université de Toulouse.

DESALLE Y., GAUME B. & DUVIGNAU K. (2009). SLAM : Solutions lexicales automatique pour métaphores. *Traitement Automatique des Langues*, **50**(1), 145–175.

DINU G. & LAPATA M. (2010). Measuring distributional similarity in context. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, p. 1162–1172, Cambridge, MA.

ERK K. & PADÓ S. (2009). Paraphrase assessment in structured vector space : Exploring parameters and datasets. In *Proceedings of the Workshop on Geometrical Models of Natural Language Semantics*, p. 57–67, Athens, Greece.

23. `http://fr.wikipedia.org/wiki/Methode_de_Monte-Carlo`

ERK K. & PADÓ S. (2010). Exemplar-Based Models for Word Meaning in Context. In *Proceedings of the ACL 2010 Conference Short Papers*, p. 92–97, Uppsala, Sweden.

C. FELLBAUM, Ed. (1998). *WordNet : An Electronic Lexical Database*. MIT Press.

GAUME B. (2004). Balades Aléatoires dans les Petits Mondes Lexicaux. *I3 : Information Interaction Intelligence*, **4**(2).

GIULIANO C., GLIOZZO A. & STRAPPARAVA C. (2007). FBK-irst : Lexical substitution task exploiting domain and syntagmatic coherence. In *Proceedings of the 4th workshop on Semantic Evaluations (SemEval-2007)*, p. 145–148, Prague, Czech Republic.

HASSAN S., CSOMAI A., BANEA C., SINHA R. & MIHALCEA R. (2007). UNT : Subfinder : Combining knowledge sources for automatic lexical substitution. In *Proceedings of the 4th workshop on Semantic Evaluations (SemEval-2007)*, p. 410–413, Prague, Czech Republic.

HAWKER T. (2007). USYD : WSD and lexical substitution using the web1t corpus. In *Proceedings of the 4th workshop on Semantic Evaluations (SemEval-2007)*, p. 446–453, Prague, Czech Republic.

LAFOURCADE M. (2007). Making People Play for Lexical Acquisition with the JeuxDeMots prototype. In *SNLP'07 : 7th Int. Symposium on NLP*, Pattaya, Thailand.

MANNING C. D., RAGHAVAN P. & SCHÜTZE H. (2008). *Introduction to Information Retrieval*. Cambridge University Press.

MARTINEZ D., KIM S. & BALDWIN T. (2007). MELB-MKB : Lexical substitution system based on relatives in context. In *Proceedings of the 4th workshop on Semantic Evaluations (SemEval-2007)*, p. 237–240, Prague, Czech Republic.

MCCARTHY D. (2002). Lexical substitution as a task for WSD evaluation. In *Proceedings of the ACL-02 Workshop on Word Sense Disambiguation : Recent Successes and Future Directions - Volume 8*, p. 109–115, Philadelphia, PA : WSD-02.

MCCARTHY D. & NAVIGLI R. (2007). SemEval-2007 Task 10 : English Lexical Substitution Task. In *Proceedings of the 4th workshop on Semantic Evaluations (SemEval-2007)*, p. 109–115, Philadelphia, PA : WSD-02.

MCCARTHY D. & NAVIGLI R. (2009). The english lexical substitution task. *Language Resources and Evaluation*, **43**, 139–159.

MOHAMMAD S., HIRST G. & RESNIK P. (2007). Tor, TorMd : Distributional profiles of concepts for unsupervised word sense disambiguation. In *Proceedings of the 4th workshop on Semantic Evaluations (SemEval-2007)*, p. 226–233, Prague, Czech Republic.

NAVARRO E. (2013). *Métrologie des graphes de terrain, application à la construction de ressources lexicales et à la recherche d'information.* PhD thesis, Université de Toulouse.

SAGOT B., CLÉMENT L., ÉRIC VILLEMONTE DE LA CLERGERIE & BOULLIER P. (2006). The Lefff 2 syntactic lexicon for French : architecture, acquisition. In *Proceedings of LREC'06*, Gênes, Italie.

SHUTOVA E. (2010). Automatic metaphor interpretation as a paraphrasing task. In *Proceedings of NAACL 2010*, Los Angeles, USA.

SHUTOVA E., VAN DE CRUYS T. & KORHONEN A. (2012). Unsepervised metaphor paraphrasing using vector space model. In *Proceedings of COLING 2012*, Mumbai, India.

THATER S., FERSTENAU H. & PINKAL M. (2010). Contextualizing semantic representations using syntactically enriched vector models. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, p. 948–957, Uppsala, Sweden.

URIELI A. (2013). *Robust French syntax analysis : reconciling statistical methods and linguistic knowledge in the Talismane toolkit*. Thése soutenue à l'université de Toulouse - école doctorale CLESCO.

VAN DE CRUYS T., POIBEAU T. & KORHONEN A. (2011). Latent Vector Weighting for Word Meaning in Context. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, p. 1012–1022, Edinburgh, UK.

YURET D. (2007). KU : Word Sense Disambiguation by Substitution. In *Proceedings of the 4th workshop on Semantic Evaluations (SemEval-2007)*, p. 207–214, Prague, Czech Republic.

ZHAO S., ZHAO L., ZHANG Y., LIU T. & LI S. (2007). HIT : Web based scoring method for English lexical substitution. In *Proceedings of the 4th workshop on Semantic Evaluations (SemEval-2007)*, p. 173–176, Prague, Czech Republic.